Decision Support for the Efficient Annotation of Bioacoustic Events

Anthony Truskinger, Michael Towsey, Paul Roe

QUT Bioacoustics
Science and Engineering Faculty
Queensland University of Technology
Brisbane, Australia

Corresponding author: Anthony Truskinger
anthony.truskinger@student.qut.edu.au
+61 7 3138 9381

S Block, Level 10, S1002
Gardens Point Campus, 2 George St, Brisbane, QLD 4000, Australia

Abstract

Acoustic sensors allow scientists to scale environmental monitoring over large spatiotemporal scales. The faunal vocalisations captured by these sensors can answer ecological questions, however, identifying these vocalisations within recorded audio is difficult: automatic recognition is currently intractable and manual recognition is slow and error prone. In this paper, a semi-automated approach to call recognition is presented. An automated decision support tool is tested that assists users in the manual annotation process. The respective strengths of human and computer analysis are used to complement one another. The tool recommends the species of an unknown vocalisation and thereby minimises the need for the memorization of a large corpus of vocalisations. In the case of a folksonomic tagging system, recommending species tags also minimises the proliferation of redundant tag categories.

We describe two algorithms: (1) a "naïve" decision support tool (16%-64% sensitivity) with efficiency of $O(n)$ but which becomes unscalable as more data is added and (2) a scalable alternative with 48% sensitivity and an efficiency of $O(\log n)$. The improved algorithm was also tested in a HTML-based annotation prototype. The result of this work is a decision support tool for annotating faunal acoustic events that may be utilised by other bioacoustics projects.

Keywords – Similarity Search, Bioacoustics, annotations, semi-automated, decision support, faunal vocalisation

1 Introduction

Acoustic sensors are an effective method for the large scale monitoring of fauna within an ecosystem. They can objectively record data over large spatiotemporal scales and the recordings can be used for ecological tasks such as determining species presence/absence. However, raw audio data is opaque – it must be analysed before it is of any use. Manual processing of audio (e.g. by having an appropriately qualified expert listen to recordings) can identify species accurately but is slow. On average, it required two minutes of listening for an expert to identify the bird species in one minute of audio (Wimmer, Towsey, Roe, & Williamson, 2013). On the other hand, automated methods, although they hold out the promise of being fast, do not have the accuracy currently required for ecological studies (Potamitis, Ntalampiras, Jahn, & Riede, 2014). There has been some success with various single-species recognisers (Brandes, Naskrecki, & Figueroa, 2006; Hu et al., 2009; Kasten, McKinley, & Gage, 2010; Towsey, Planitz, Nantes, Wimmer, & Roe, 2012; Wimmer, Towsey, Planitz, Williamson, & Roe, 2013) and some multi-species recognisers (Acevedo, Corrada-Bravo, Corrada-Bravo, Villanueva-Rivera, & Aide, 2009; Anderson, Dave, & Margoliash, 1996; Harma, 2003). A state-of-the-art recogniser has been reported by Stowell and Plumbley (2014) but its application to the pre-prepared *lifeclef2014* dataset (Joly et al., 2014) does not necessarily translate to the rigorous requirements of an ecological study. Even capable automatic recognisers often require manual verification (Wimmer, Towsey, Planitz, et al., 2013).

An alternative analytical approach for recordings of faunal vocalisations is to extract ecological indices which point to the presence of animal vocalisations of interest rather than identifying the actual species (Bart, 2005; Depraetere et al., 2011; Gage, Napoletano, & Cooper, 2001; Gasc et al., 2013; Pieretti, Farina, & Morri, 2011; Towsey, Wimmer, Williamson, & Roe, 2014). This approach is part of the emerging field of *soundscape ecology* that views the acoustic world from an ecological perspective rather than a species perspective (Pijanowski, Farina, Gage, Dumyahn, & Krause, 2011).

Humans can become excellent classifiers of bioacoustic events given sufficient training but manual analysis of audio data is a laborious process, the more so for experts. It is also expensive. However,

humans can work more efficiently if given appropriate technical support. This so-called *semi-automated* approach combines the complementary strengths of human and computer. In this paper, we explore a *semi-automated* approach to the identification of animal vocalisations, primarily but not exclusively due to birds. When annotating, users are given a short sample of audio and its pictorial representation as a spectrogram; the user is required to identify the species making the call. Decision support takes the form of a "suggestion tool" that shows similar *labelled* samples of audio and spectrograms to the user. We have previously reported a proof-of-concept decision support system embedded in a website (Truskinger et al., 2011). The purpose of that paper was to test the effectiveness of the suggestion tool on the performance of a mix of expert and non-expert participants. The authors report a slight (but statistically significant) increase in the participant classification rate but not in their classification accuracy. Interviews with participants indicated that the suggestion tool was potentially helpful but needed to be more accurate. The participant feedback provides the motivation for the work described in this paper.

The suggestion tool reported by Truskinger et al. (2011) relied on 400 *reference annotations*. A reference annotation is one determined by experts as being a good exemplar of its class. In this paper, we investigate the hypothesis that a decision-support system dependent on typical annotations (as opposed to exemplars) would improve in accuracy. The remainder of this paper is organised as follows: Section 2 describes related work. Sections 3, 4 and 5 describe our methodology, results, and discussion respectively. The final sections describe future work and conclude.

2 Related Work

Annotating multimedia data with tags is a common practice on the web. Examples of multimedia annotation include: Flickr (images), SoundCloud (sound), YouTube (video formats), and Vannotea (Schroeter, Hunter, Guerin, Khan, & Henderson, 2006) which can annotate most multimedia formats. This research focuses on annotating audio data for ecological science. Similar research projects have cultivated libraries of audio recordings that have been labelled (usually the entire recording is labelled). The 'Jacques Vielliard' dataset maintained by UNICAMP (Cugler, Medeiros, & Toledo, 2011) and the Berlin Sound Archive (Bardeli, 2009) are two examples. These libraries are excellent resources; however, the majority of their recordings are not acoustic sensor recordings. Instead, they are usually targeted and have high *signal-to-noise ratios* (SNRs).

Analysts, including novices, find the detection and isolation of bioacoustic events from background events to be easier than the classification of those events. Because a large corpus of audio patterns must be memorised in order to classify events, few people have enough experience or skill to identify all faunal vocalisations by recall alone. Even a geographically constrained set of recordings from just one *site* can contain hundreds of vocalising species. Some of these species, especially birds, have more than one form of vocalisation. For example, at QUT's SERF facility, located in the Samford Valley, Queensland, Australia, 460 unique tags (*classes*) have been applied to 100 species, found in 80 000 bioacoustic events from six days of data (Wimmer, Towsey, Planitz, et al., 2013).

Some experts can aurally classify large numbers of bird species by recall alone. These experts have had many years of training as ornithologists or through recreational *birding* activities. However, their memorised knowledge is limited to the geographical areas where they have had experience; different environments often mean different sets of species. Vocalisations of species can also vary between regions creating further difficulty (Kirschel et al., 2009).

Nevertheless, humans are exceptional at pattern recognition tasks (Sroka & Braida, 2005) and identification of acoustic events becomes easier when a spectrogram accompanies the audio data (Wimmer, Towsey, Planitz, et al., 2013). Most analysts can discern visual differences between spectrogram features with ease. Human selected discriminating features are creative, often qualitative, and describe aspects of an object that are hard to quantify (Feyyad, 1996). Humans can discriminate audio-patterns even in noisy, degraded, or overlapping signals (Rusu & Govindaraju, 2004). To summarize, any method to augment the skills of human analysts should utilise their exceptional comparison skills and place less emphasis on recall of prior knowledge.

A decision support tool for bioacoustic events imposes a set of constraints on the *user interface* (UI). The autocomplete box, a similar but far less complex UI mechanism, suggests possible textual matches within milliseconds, sometimes from remote sources. Likewise, an effective decision support tool must also provide results in sub-second times as its utility depends on its response-time. The recommended response-time for page navigation is sub-second and for interactive visual components is less (Miller, 1968; Nielsen, 1999).

The task of matching a 'sound-bite' to a larger database of audio for the purpose of classification has been previously accomplished in both the ecological acoustics and music fields (Bardeli, 2009; Kasten, Gage, Fox, & Joo, 2012; Wang, 2006). Because vocalisations occur in noisy environments and vary greatly by region, music matching methods are ineffective for matching faunal vocalisations (Cugler et al., 2011). Currently, there is no effective system for automated content-based similarity search of faunal vocalisations. The existing partial-solutions to similarity search all require signal processing to extract features and complex classification algorithms. Given the immense volume of data collected by acoustic-sensors, the difficulty of the classification task, and the need to generate suggestions quickly, the suggestion task lends itself to a metadata-based solution.

Other sound ecology software packages have been created that may benefit from the approaches in this paper. The Pumilio project is an open source software package that allows researchers to store audio recordings (Villanueva-Rivera & Pijanowski, 2012). Pumilio allows recordings to be uploaded, analysed, and tagged with metadata, through a web interface. Similarly, The REAL digital library is an archive of sensor recordings accessible through a web interface. The REAL project also allows automated analysis and has search capabilities (Kasten et al., 2012).

3 Experimental method

Increasing the quantity of training data is a standard approach used to increase the accuracy of supervised machine-learning problems (Zhu, Vondrick, Ramanan, & Fowlkes, 2012). We have previously published results for an experiment where the training data consisted of 400 exemplar annotations; that is, the canonical or best examples of calls for each class. However, most ordinary acoustic events in real recordings of the environment are distorted by noise or overlapping events. Furthermore, the majority of recorded vocalisations have low signal-to-noise ratios. Low SNR is seen as an effect of the combination of the inverse-square law and the probable distribution of fauna around a sensor; it is likely that more vocalising individuals will be further from the microphone. In this work, we investigate the hypothesis that increasing the proportion of poorer quality calls (relative to high SNR canonical calls) within the training data will increase the accuracy of the resulting decision support tool.

A large increase in the quantity of training data affects the choice of algorithmic approach (Deng, Berg, Li, & Fei-Fei, 2010). For the decision support tool, new algorithms are tested for their scalability and ease of implementation. To achieve scalability, the feature set was kept to a minimum. In

particular, we focused on easy-to-extract features derived from the meta-data of an annotated call as opposed to audio-content features.

The experimental framework for this research was to evaluate performance for multiple simulations of the decision support tool over different combinations of datasets, algorithmic components, and feature sets. This section describes the components of the simulations.

3.1 Datasets

Two datasets were used for the experiment: the *Full* dataset and the *Reference* dataset. Both datasets use the same testing data. Table 1 has a summary breakdown on the number of annotations and their tags, for each dataset.

Table 1 – The annotations (instances) and tags (classes) in each dataset

Dataset	Number of Annotations (bounded events)		Number of Tags (unique call labels)		Tags Present in Both Training and Test
	Training	Test	Training	Test	
Full SERF Dataset	60 746	21 035	382	207	116
Reference Library Dataset	434		327		180

The *Full* dataset consists of annotations generated by human analysts, in audio recordings taken from the QUT Samford Ecological Reserve Facility (SERF), located north-west of Brisbane, Queensland, Australia. The annotated dataset was produced by Wimmer, Towsey, Planitz, et al. (2013). The vegetation at SERF is mainly open-forest to woodland comprised primarily of *Eucalyptus tereticornis*, *Eucalyptus crebra* and *Melaleuca quinquenervia* in moist drainage. There are also small areas of gallery rainforest with *Waterhousea floribunda* predominantly fringing the Samford Creek to the west of the property, and areas of open pasture along the southern border. Faunal vocalisations were analysed by experts producing 473 call types (tags) for 96 species across four sites. The majority of the species identified were *Aves;* however, there are examples of crickets, frogs, and marsupials in the dataset. The most frequently detected species include the Rufous Whistler (*Pachycephala rufiventris*), Lewin's Honeyeater (*Meliphaga lewinii*), Torresian Crow (*Corvus orru*), Olive-backed oriole (*Oriolus sagittatus*), and Scarlet Honeyeater (*Myzomela sanguinolenta*). The least detected species include the Pale-vented Bush-hen (*Amaurornis moluccana*), Glossy Black Cockatoo (*Calyptorhynchus lathami*), Forest Kingfisher (*Todiramphus macleayii*), Collared Sparrowhawk (*Accipiter cirrhocephalus*), and Azure Kingfisher (*Alcedo azurea*).

The training data (Table 2) consists of annotations taken from three sites separated approximately 300m apart. The testing data consists of annotations taken from the fourth site. The data was partitioned this way to simulate a real system state – i.e. users annotate some sites first generating instances that can be used later for the decision support of other sites.

Table 2 – The source sites within SERF that are used in the Full dataset

Data Type	Site	Date Range	Days
Training	North West	13th-24th of October 2010	12
	South East	13th-24th of October 2010	12
	South West	13th-18th of October 2010	6
Test	North East	13th-18th of October 2010	12

The *Reference* dataset was designed to emulate the training data described in our original implementation of the suggestion tool (Truskinger et al., 2011) and is a copy of the reference annotation library extracted in 2011. This dataset consists of annotations that experts have marked as good quality exemplars of their class. Analysts use them as a standard reference for new annotation work. The reference annotations are geographically well distributed and come from Brisbane, St. Bees Island (off Queensland's central coast), and other locations throughout Australia. This dataset is composed of a broad range of species. Notably, there are more examples of non-Avian species like Koalas (*Phascolarctos cinereus*) and Canetoads (*Rhinella marina*).

3.2 Features

Each annotation identifies a section of audio tagged with a label, bounded in the frequency and time domains. Ideally, each annotation should encompass one acoustic event but most contain background noise and/or overlapping signals from other sources. Fig 1. depicts the basic properties of an annotation.

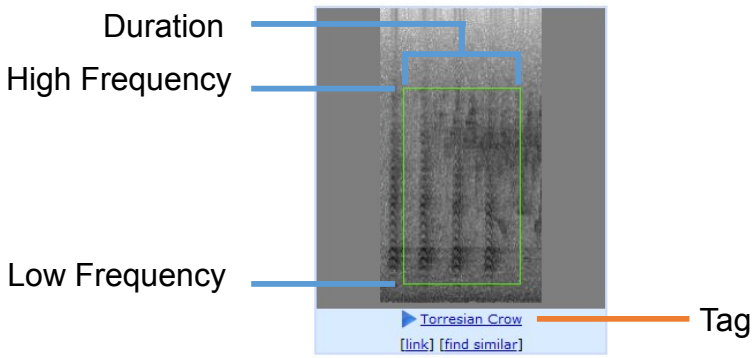

Figure 1 – A diagram of the three bounding features of an annotation. The acoustic event that has been annotated is a two-second, broadband bark of a Torresian Crow (Corvus orru)

The three main features used for this study are the *low frequency bound*, the *high frequency bound*, and *time duration*. These three 'bounding' features are continuous in value (as opposed to discrete) and were chosen due to their compact format, easy availability, and almost zero computational requirements. These features are determined by human action and are not derived computationally from the underlying audio or spectrogram. Accordingly, the features are noisy because the annotators were not consistent within or between themselves in how they placed a bounding box in relation to the enclosed acoustic event.

A final feature is the *tag*: a class label that defines the content of the bounding box. The tag labels are used as the output of the decision support tool. The tags are textual and suffer from numerous spelling and grammatical errors, suffixes that conflate their semantics, and various forms of synonymy (Truskinger et al., 2013). The tag 'unknown' allowed an additional class to accommodate analyst uncertainty. The tag data was cleaned manually for this experiment.

3.3 Algorithm Components

A typical decision support tool presents the user with a limited number of choices that are contextually meaningful. From an algorithmic perspective, the output of the tool is a list of suggestions ranked by a similarity metric.

3.3.1 Similarity Search

The simplest similarity metric is the Euclidean distance as used by Truskinger et al. (2011):

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (\boldsymbol{q}_i - \boldsymbol{p}_i)^2}$$

Equation 1– The Eucldiean distance formula

Its computational complexity is linear in size of the feature set and training set, rendering it unsuitable for large datasets.

3.3.2 Scale Reduction with Prototypes

The computational complexity of a Nearest Neighbour algorithm with 60 000 training instances is prohibitive. It would be desirable to reduce overall training set size whilst maintaining similar decision support performance. As shown in Table 1, the 60 000 training instances encompass 382 classes (tags). We grouped and calculated the centroid for each class to produce a reduced set of 382 prototypical instances.

3.3.3 Normalization

The Euclidean distance metric performs poorly if the features differ significantly in scale. Normalising input features using the z-score function (Equation 2) compensates for this problem.

$$z = \frac{x - \mu}{\sigma}$$

Equation 2 - The z-score

In our work, there are two ways to apply the z-score transformation: (1) the z-score can be calculated separately for each class whilst calculating class centroids (see section 3.3.2); or (2) the z-score normalization can be calculated over the entire dataset of 60 000 annotations prior to calculating the class centroids. Method (1) normalises within-class groups producing greater variances in the transformation and thus making a better discriminator. However, the z-score is undefined for single instance classes (this occurs for 11 classes) which must therefore be ignored. Method (2) yields less variance for each feature, however, the mean and standard deviations are generally better defined due to the higher number of instances in the populations. Both methods of normalization have similar computational cost. Additionally, test set features must be normalised using means and standard deviations derived from the training data.

3.3.4 Randomization Protocol

A randomization protocol was used to assess baseline performance for combinations of algorithms. A successful method must perform significantly better than randomised trials. The randomization protocol was implemented by randomly reassigning the tags on annotations. The Fisher-Yates shuffle (Knuth, 1969) was used to randomise training data for every experimental trial configuration and applied after basic cleaning of the data but before remaining steps in the algorithm. Note that only the class labels were shuffled (not the other instance features) to ensure that valid feature vectors were used.

3.3.5 Algorithm Combinations

Simulations were run to determine the difference in performance for every combination of algorithm, feature, and dataset. The algorithmic workflow has three varying components: class labels randomisation (or not), data normalisation (two methods, or not), and grouping instances into prototypes (or not). The three bounding box features were tested with all their combinations ($3! = 8$). Thus, the 12 algorithm combinations and 8 feature combinations produced 96 combinations (simulations) per dataset. Each simulation tests each *test instance* from the *test* portion of a dataset.

3.4 Evaluation Method

We evaluate the response to a *test instance* from the rank of the first correct instance in the first *N* instances of the returned output list. For a given value of *N*, the output list can be considered a true positive if the first correct response is ranked at $r \leq N$. *N* is a threshold rank taking values from 1 to the number of training instances in the training model. Conversely, the list is a false negative if the first correct response is ranked at $r > N$.

3.4.1 Sensitivity and Accuracy

Performance on test data was evaluated for sensitivity and accuracy. For a given value of *N*, accuracy (A) is defined as:

$$A = \frac{P}{T}$$

where P is the number of true positives returned in the first *N* output instances and T is the total number of queries in the test data. Because the test data includes some instances whose class is not found in the training data (and therefore can never be identified), we also calculate, for a given value of *N*, the sensitivity (S) defined as:

$$S = \frac{P}{P_{total}}$$

where P_{total} is the total number of test instances whose class is also found in the training data of the Nearest Neighbour model. Note that these definitions are not the usual definitions applied to performance of binary classifiers because the concept of false positive and true negative is not defined in the context of the decision support task. For this reason, summary performance curves such as the Receiver-Operator Characteristic (ROC) curve (Balakrishnan, 1991), are also not appropriate for this task. Instead, we adopt Sensitivity Response Curves.

3.4.2 Sensitivity Response Curves

Using the above measure of sensitivity, the area under a plot of sensitivity versus N is a useful summary measure of decision support performance. We call these plots *sensitivity response graphs* and the area under the curve (AUC) is given by:

$$AUC\ Score = \frac{1}{2}\sum_{n=1}^{N}(S_n + S_{n+1}) = \sum S$$

This formula uses the standard trapezoidal rule for approximating the area under a curve. The AUC score is normalized by the size of the dataset as follows:

$$AUC\ Score\ (normalised) = \frac{\sum S}{N}$$

which evaluates the area under the sensitivity response curve relative to a perfect response curve. The normalised AUC score for random trials is expected to be closer to 0.5. The closer the normalised AUC score is to 1.0, the better the performance of the suggestion algorithm.

4 Results

All of the 192 dataset/analysis combinations were tested but only the most relevant findings are reported. A comparison between the new and previously published results is presented along with computational performance comparisons. All results presented have three features: Start Frequency, End Frequency, and Duration – using fewer features did not perform as well. Other

features and combinations were tested, including a *Time of Day* feature (both angular similarity and phase of day). These other features did not perform well and their results are omitted. Both methods of z-score normalisation were compared in the experiments. However, the 'global' normalisation did not perform well and results are presented only for within-class normalisation. The full set of results can be obtained by contacting the author.

4.1 Simulation Performance

The decision support tool's performance for this experiment is summarised in Table 3, and Figs. 2 and 3. Table 3 lists the properties of the simulations reported, including dataset, algorithmic components, sensitivity for five results, and time taken to return five results. We empirically determined that showing five results fits well into a variety of user interfaces. As such, when presenting results, statistics for showing the top five suggestions are presented along with AUC scores.

Table 3 – Reported Simulations, their Composition, and Performance

Simulation	Dataset	Name	Description	AUC	Sensitivity (N=5)	Time Taken (s, N=5)	Ranks shown N/n
1	Reference	Reference-Basic	Similarity search only	0.76	0.2456	0.025	380/434
2	Full	Full-Basic	Similarity search only	0.89	0.6408	3.200	380/60 746
3	Full	Full-Prototypes	Prototypes, Normalised, Similarity search	0.97	0.4812	0.055	380/383
4	Full	Full-Random-Prototypes	Randomised, Prototypes, Normalised, Similarity search	0.78	0.0070 ± 0.0175	0.062	380/383

The graph in **Error! Reference source not found.**Figure 2 shows the computation time and memory usage required *per result* (test case). The time taken statistics are multiplied by five to produce *time taken per query*. These statistics were derived from the log files of the simulations.

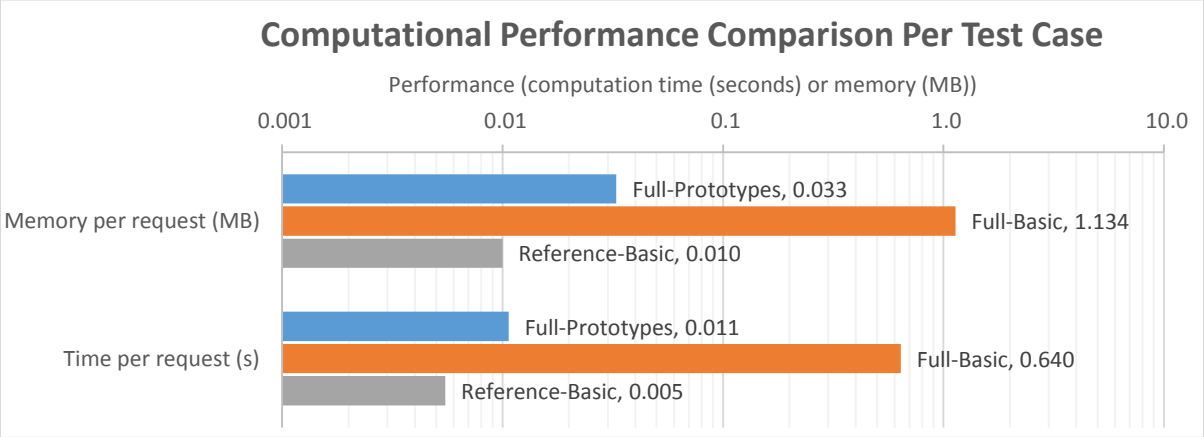

Figure 2 – Computational performance differences for the simulations. Reported are the time and memory requirements needed to produce one result (a test case).

The sensitivity response graphs in Figure 3 (over page) provide the performance profile of an analysis across all ranks. The inset of Figure 3 summarises the results for low ranks – which are equivalent to showing limited suggestions in a user interface.

4.1.1 Reference Dataset, Basic Algorithm

The Reference-Basic simulation used the Reference dataset with a basic similarity search only. This result represents the effectiveness of the decision support tool as it was implemented by Truskinger et al. (2011). The results for this simulation are in row 1 of Table 3.

4.1.2 Full Dataset, Basic Algorithm

The Full-Basic simulation used the Full dataset with a basic similarity search only. The results listed for this analysis are the outcome of adding more training data without improving the algorithm. The results for this simulation are in row 2 of Table 3. Performance does improve substantially when more training data is added; the AUC score increases to 0.89 (Full-Basic) from 0.76 (Reference-Basic) and the correct suggestions within the top-five probability increases to 64.08% from 24.56%, Δ +40.52%. However, for time taken to generate a query, there are two orders of magnitude in difference between the Reference-Basic and the Full-Basic simulations: 3.2s and 25ms of CPU time respectively.

4.1.3 Full Dataset, Prototype Algorithm

The Full-Prototypes simulation used the Full dataset. The algorithm used created class prototypes and normalized instances within their groups before conducting the similarity search. The results for this simulation are in row 3 of Table 3. The Full-Prototypes simulation produced an AUC score of 0.97, an improvement over the Reference-Basic simulation (0.78). The top-five statistic for the Full-Prototypes simulation is 48.12%; compared to the Reference-Basic result of 24.56% there was 23.56% improvement. However, the Full-Prototypes simulation does not perform as well as the Full-Basic simulation: 48.12% compared to 64.08% (Δ -15.96%). For time taken to generate a query, the Full-Prototypes simulation completed in 55ms; that is two orders of magnitude quicker than the Full-Basic simulation's 3.2s.

The randomised simulation (Full-Random-Prototypes) is reported in conjunction with the Full-Prototypes simulation to demonstrate comparative baseline performance. The algorithm and dataset are identical to the Full-Prototypes simulation and the results are in row 4 of Table 3. The Full-Random-Prototypes result is stochastic (unlike the other deterministic simulations); as such, the simulation was run 100 times and the mean values are used to graph and calculate AUC. Additionally, one standard deviation is shown on the Full-Random-Prototypes series in Figure 3 as ‘error’. The AUC score for the Full-Random-Prototypes simulation is 0.78 – which is substantially lower than the non-random 0.97 AUC score. The full effect can be seen in Figure 3.

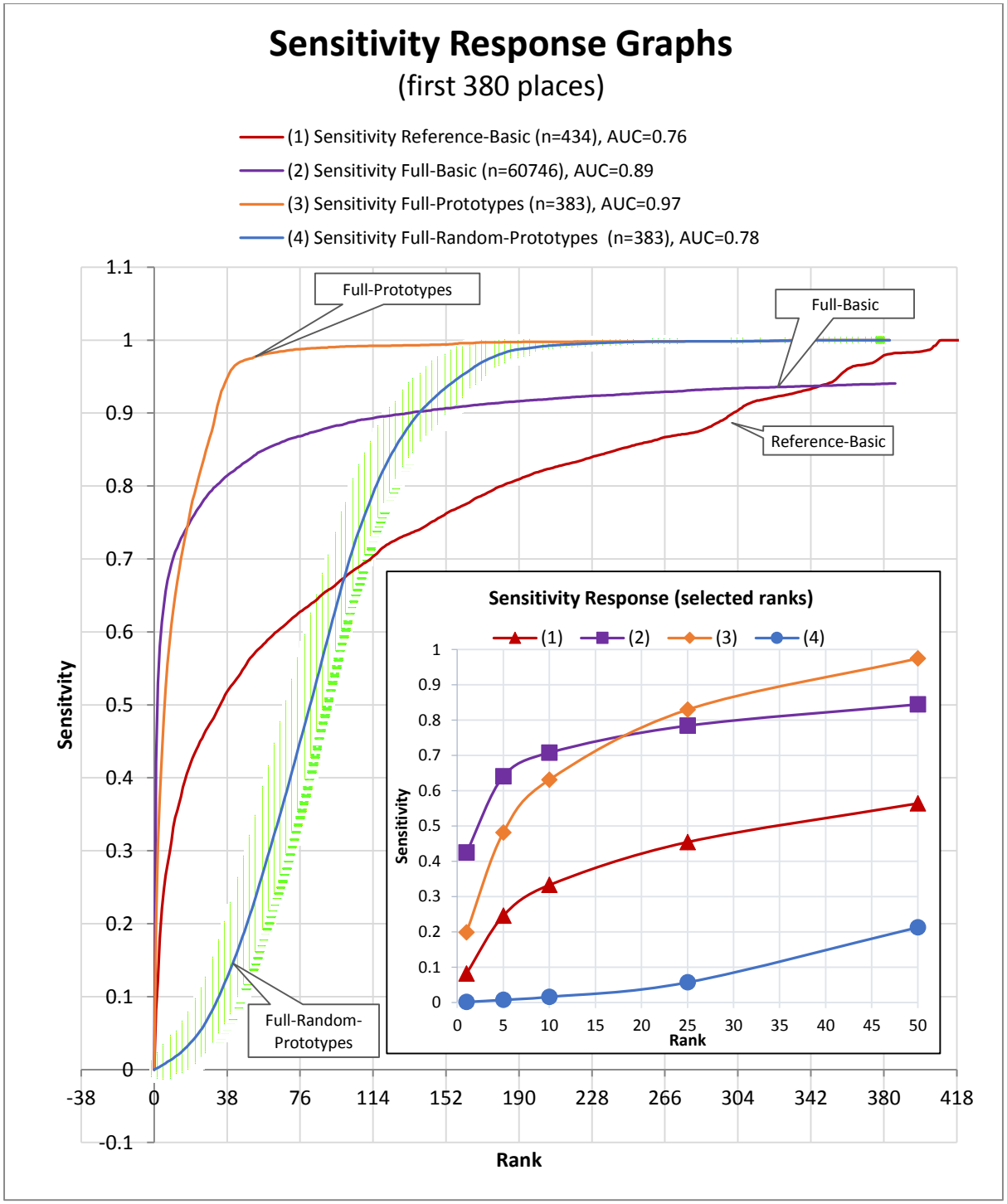

Figure 3 – Sensitivity response curves for the analyses. Inset: Sensitivity response curves for low ranks.

4.2 Design Proof of Concept

After the experiment, the Full-Prototypes simulation was used in a specialised interface designed for supporting the annotation decisions of a user. This new prototype (compared to the UI shown by Truskinger et al. (2011)) was designed to be easier to use and more focused on annotation with a specialised space for decision support suggestions.

The prototype demonstrates how the performance improvements would practically work in a website. The prototype (see Fig. 4**Error! Reference source not found.**) was implemented on a test website; all features shown are functional, including the weather statistics, the location statistics, the satellite imagery link, the decision support tool, and the multi-tagging fields.

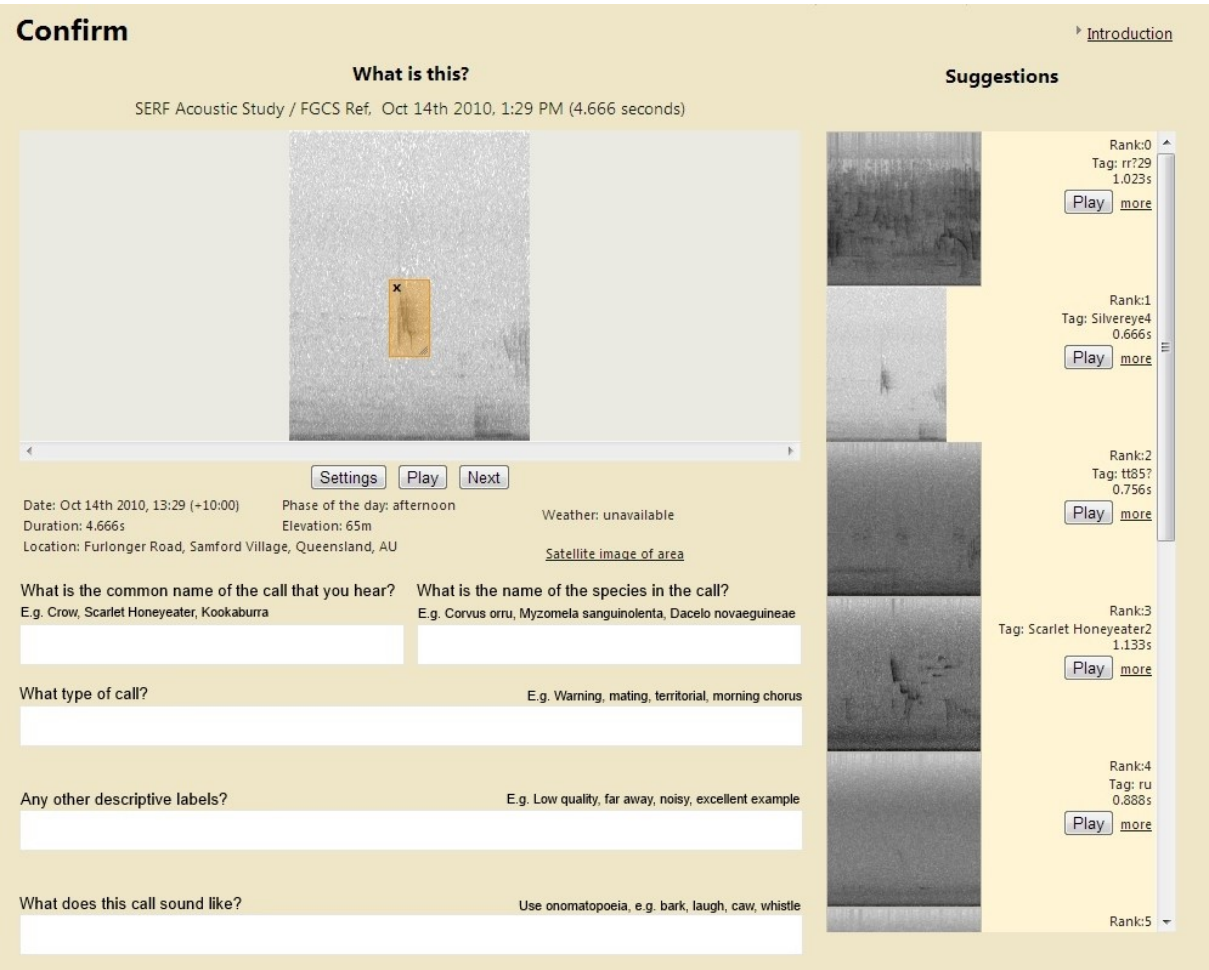

Figure 4 – A screenshot of a prototype user interface with the decision support tool built in

The main sections of the prototype are the spectrogram area, the overlayed annotation drawing surface, and the suggestion list shown on the right hand side. The screenshot shows the state of the tool, after an unknown event had been shown on the screen, after the bounding box has been drawn, and after the decision support suggestions had been returned. The next tasks are tagging and then saving the annotation. The correct result, a *Silvereye4* is shown as the second suggestion. The *Silvereye4* tag is the folksonomic label applied to this call type – it is a concatenation of the common name and a numeral suffix indicating the call type.

5 Discussion

Producing datasets of faunal vocalisations for ecologists is complex and time consuming. The aim of a decision support tool is to aid the human analysts generating this data (be they ecologists, citizen scientists, or graduate students). This research sought to improve the performance of a decision support tool whilst requiring that suggestions were returned in a timely manner for an interactive scenario. It was hypothesised that adding ordinary annotations into the training set would improve performance.

Adding more training data and using just the Euclidean metric for similarity used in the previous suggestion tool paper (Truskinger et al., 2011) are computationally inefficient. To confirm this, the experiments that were run included subjecting the test data to both the training dataset from the original paper and the improved Full dataset. Performance improved considerably between the Reference-Basic and Full-Basic simulations, satisfying the hypothesis that adding ordinary annotations will increase decision support performance. The training data added (60 476 annotations) for the experiment is a fraction of the data now available; at the time of writing[1], there were approximately 70 000 annotations that could be used as additional training data.

The Full-Basic simulation is an ideal result if computational performance were irrelevant. Since performance is relevant, the 3.2s of 100% CPU utilisation (row 2, Table 3) is too slow for a responsive user interface. Additionally, there is no excess time for the web server to complete the other tasks required for returning each decision support query to a client (meta-data retrieval, spectrogram generation, and audio segment cutting for showing suggestions). With more users or more training data — both readily available in a real world system — the computational performance of this method will decline linearly for both factors (that is $O(n)$).

The Full-Basic simulation was an experiment conducted to formally test the assumption that the method would not scale. To find a better solution, other simulations were trialled. We believe that the best result was the Full-Prototypes simulation.

The Full-Prototypes simulation performs twice as well (48% vs 24%) as the original methodology (Reference-Basic) but is not as effective as the Full-Basic simulation (64%). However once the training is completed for the Prototypes algorithm, it is far more computationally efficient. The Full-Prototypes method can return a set of five suggestions to a user in 55ms (compared to the basic similarity search requiring 3.2s for the same data).

In summary, the additional algorithmic components in the Full-Prototypes simulation offers a substantial improvement in decision support performance and at 55ms for five suggestions, falls well within the sub-second requirements for a decision support tool – with time to spare for extracting the additional audiovisual data that is required for displaying suggestions.

The manner in which the Full-Prototypes algorithm scales is significant. Both algorithms (the basic similarity search and the more complex Prototypes algorithm) will suffer from a linear performance drop with more users. However, Prototypes algorithm will not scale linearly when more training data is added. For example, with double the training data, the Basic algorithm will be twice as slow (200%, an estimated 6.4s per query). This is coincides with an $O(n)$ efficiency profile. As the Prototypes algorithm groups common tag types into prototypes, performance will only decrease if new tag types (classes) are added. In our experience, the vast majority of annotations share common tags and new tags are rare. In the additional data that can be added to the system (another 70 000 annotations currently available that were not included in this paper), there are only 100 additional unique tags, resulting in an estimated 20% increase in required computational resources (120%, an estimated 66ms per query). The rate of classes added per training instance added coincides with an $O(\log n)$ efficiency profile ($R^2 = 0.81$).

In practical terms, this decision support tool can be easily extended and applied to similar systems. The methods presented in this paper are generalized: for any bioacoustics project, where annotation of acoustic events within recordings (continuous or sampled) is needed, this method of decision support could be applied. For example, the authors have used and inspected the Pumilio software

[1] July 2014. Updated statistics can be requested from the author.

package (Villanueva-Rivera & Pijanowski, 2012) – this decision support method can be implemented within that project. Further, the presented algorithm essentially relies on the basic acoustic properties of a faunal vocalisation (time and frequency bounds) to discriminate between them. Despite this experimental dataset containing mostly avian vocalisations, this method is theoretically general enough to be reused for the decision support of other species.

6 Future Work

This research focused on a few simple algorithmic components that were rigorously tested. However, other algorithmic components also have potential. Examples include sub-clustering within groups, Bayesian classifiers, and [random forest] decision trees. Using a *k*-nearest neighbour classifier (instead of Euclidean similarity) is estimated to reduce computation by up to 80%.

In particular, sub-clustering within prototypical groups is an enhancement expected to increase suggestion performance with only a moderate trade off in computational performance. It is evident that certain class prototypes exhibit a large variability as a bimodal distribution in one or two features – this effect is seen commonly with the end frequency feature. If these cases can be split into distinct prototypes by clustering the sub-distributions, it will decrease the negative impact that calculating centroids has when forming prototypes.

Other audio and metadata features, have the potential to improve suggestion performance. Contextual features, like location, weather patterns, and seasonal variations, describe the environment in which the audio was recorded. These features were investigated but did not exhibit any significant variance for the limited spatiotemporal scale of the available data. Annotations collected from larger spatiotemporal context may exhibit a greater ability to distinguish annotations.

Fauna often vocalise according to diurnal patterns and as such, we implemented two concepts for measuring the similarity between times of the day: *angular similarity* and the *phase of day*. Neither feature performed well in testing. The poor performance of the time of day feature was unexpected as there is evidence from ecological and citizen experts alike that the time of day of a vocalisation is produced is a useful distinguishing feature – more investigation is needed.

Lastly, extending this study to fauna and datasets outside of Australia is necessary to determine the effectiveness of this decision support tool for larger scale applications. The underlying relationship between an annotation's features and the acoustic events may not exist in other environments.

7 Conclusion

This paper has presented a method for fully implementing a decision support tool. Using the presented Full-Prototypes method, the tool's suggestion performance for the top five results, in 21 035 test cases, has been increased to 48% sensitivity from 24%. Further, given the large amount of training data used, the improved algorithm scales far better in computational performance than the original algorithm ($O(\log n)$ versus $O(n)$). The newer algorithm responds with sub-second response times and has been demonstrated working in a web-based prototype. The described methodologies can potentially be applied to other bioacoustics software packages.

Acknowledgements

We acknowledge the contribution of Mark Cottman-Fields (QUT) and Jason Wimmer (QUT) in managing and creating the datasets used. We also thank the experts that helped annotate the data and in particular the efforts of Tom Tarrant.

All funding for this research was provided by the Queensland University of Technology.

References

Acevedo, M. A., Corrada-Bravo, C. J., Corrada-Bravo, H., Villanueva-Rivera, L. J., & Aide, T. M. (2009). Automated classification of bird and amphibian calls using machine learning: A comparison of methods. *Ecological Informatics, 4*(4), 206-214. doi: 10.1016/j.ecoinf.2009.06.005

Anderson, S., Dave, A., & Margoliash, D. (1996). Template-based automatic recognition of birdsong syllables from continuous recordings. *Journal of the Acoustical Society of America, 100*(2), 1209-1219.

Balakrishnan, N. (1991). *Handbook of the logistic distribution*: CRC Press.

Bardeli, R. (2009). Similarity Search in Animal Sound Databases. *Multimedia, IEEE Transactions on, 11*(1), 68-76. doi: 10.1109/TMM.2008.2008920

Bart, J. (2005). Monitoring the abundance of bird populations. *Auk (American Ornithologists Union), 122*(1), 15-25.

Brandes, T. S., Naskrecki, P., & Figueroa, H. K. (2006). Using image processing to detect and classify narrow-band cricket and frog calls. *The Journal of the Acoustical Society of America, 120*, 2950.

Cugler, D. C., Medeiros, C. B., & Toledo, L. F. (2011). *Managing animal sounds-some challenges and research directions.* Paper presented at the Proceedings V eScience Workshop-XXXI Brazilian Computer Society Conference.

Deng, J., Berg, A. C., Li, K., & Fei-Fei, L. (2010). What does classifying more than 10,000 image categories tell us? *Computer Vision–ECCV 2010* (pp. 71-84): Springer Berlin Heidelberg.

Depraetere, M., Pavoine, S., Jiguet, F., Gasc, A., Duvail, S., & Sueur, J. (2011). Monitoring animal diversity using acoustic indices: Implementation in a temperate woodland. *Ecological Indicators, In Press, Corrected Proof*. doi: 10.1016/j.ecolind.2011.05.006

Feyyad, U. M. (1996). Data mining and knowledge discovery: making sense out of data. *IEEE Expert, 11*(5), 20-25. doi: 10.1109/64.539013

Gage, S. H., Napoletano, B. M., & Cooper, M. C. (2001). Assessment of ecosystem biodiversity by acoustic diversity indices. *The Journal of the Acoustical Society of America, 109*(5), 2430-2430.

Gasc, A., Sueur, J., Jiguet, F., Devictor, V., Grandcolas, P., Burrow, C., . . . Pavoine, S. (2013). Assessing biodiversity with sound: Do acoustic diversity indices reflect phylogenetic and functional diversities of bird communities? *Ecological Indicators, 25*(0), 279-287. doi: http://dx.doi.org/10.1016/j.ecolind.2012.10.009

Harma, A. (2003, 6-10 April 2003). *Automatic identification of bird species based on sinusoidal modeling of syllables.* Paper presented at the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03).

Hu, W., Bulusu, N., Chou, C. T., Jha, S., Taylor, A., & Tran, V. N. (2009). Design and evaluation of a hybrid sensor network for cane toad monitoring. *ACM Trans. Sen. Netw., 5*(1), 1-28. doi: 10.1145/1464420.1464424

Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W.-P., . . . Müller, H. (2014). Lifeclef 2014: multimedia life species identification challenges *Information Access Evaluation. Multilinguality, Multimodality, and Interaction* (pp. 229-249): Springer.

Kasten, E. P., Gage, S. H., Fox, J., & Joo, W. (2012). The remote environmental assessment laboratory's acoustic library: An archive for studying soundscape ecology. *Ecological Informatics, 12*(0), 50-67. doi: http://dx.doi.org/10.1016/j.ecoinf.2012.08.001

Kasten, E. P., McKinley, P. K., & Gage, S. H. (2010). Ensemble extraction for classification and detection of bird species. *Ecological Informatics, 5*(3), 153-166. doi: http://dx.doi.org/10.1016/j.ecoinf.2010.02.003

Kirschel, A. N. G., Blumstein, D. T., Cohen, R. E., Buermann, W., Smith, T. B., & Slabbekoorn, H. (2009). Birdsong tuned to the environment: green hylia song varies with elevation, tree cover, and noise. *Behavioral Ecology, 20*(5), 1089-1095. doi: 10.1093/beheco/arp101

Knuth, D. E. (1969). Seminumerical Algorithms, volume 2 of The Art of Computer Programming. Addision Wesley. *Reading, MA*.

Miller, R. B. (1968). *Response time in man-computer conversational transactions.* Paper presented at the Proceedings of the December 9-11, 1968, fall joint computer conference, part I.

Nielsen, J. (1999). User interface directions for the Web. *Commun. ACM, 42*(1), 65-72. doi: 10.1145/291469.291470

Pieretti, N., Farina, A., & Morri, D. (2011). A new methodology to infer the singing activity of an avian community: the Acoustic Complexity Index (ACI). *Ecological Indicators, 11*(3), 868-873.

Pijanowski, B. C., Farina, A., Gage, S. H., Dumyahn, S. L., & Krause, B. L. (2011). What is soundscape ecology? An introduction and overview of an emerging new science. *Landscape Ecology, 26*(9), 1213-1232. doi: 10.1007/s10980-011-9600-8

Potamitis, I., Ntalampiras, S., Jahn, O., & Riede, K. (2014). Automatic bird sound detection in long real-field recordings: Applications and tools. *Applied Acoustics, 80*(0), 1-9. doi: http://dx.doi.org/10.1016/j.apacoust.2014.01.001

Rusu, A., & Govindaraju, V. (2004, 26-29 Oct. 2004). *Handwritten CAPTCHA: using the difference in the abilities of humans and machines in reading handwritten words.* Paper presented at the Frontiers in Handwriting Recognition, 2004. IWFHR-9 2004. Ninth International Workshop on.

Schroeter, R., Hunter, J., Guerin, J., Khan, I., & Henderson, M. (2006, Dec. 2006). *A Synchronous Multimedia Annotation System for Secure Collaboratories.* Paper presented at the Second IEEE International Conference on e-Science and Grid Computing.

Sroka, J. J., & Braida, L. D. (2005). Human and machine consonant recognition. *Speech Communication, 45*(4), 401-423. doi: http://dx.doi.org/10.1016/j.specom.2004.11.009

Stowell, D., & Plumbley, M. D. (2014). Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning. *PeerJ, 2*, e488. doi: 10.7717/peerj.488

Towsey, M., Planitz, B., Nantes, A., Wimmer, J., & Roe, P. (2012). A toolbox for animal call recognition. *Bioacoustics, 21*(2), 107-125. doi: 10.1080/09524622.2011.648753

Towsey, M., Wimmer, J., Williamson, I., & Roe, P. (2014). The use of acoustic indices to determine avian species richness in audio-recordings of the environment. *Ecological Informatics, 21*(0), 110-119. doi: http://dx.doi.org/10.1016/j.ecoinf.2013.11.007

Truskinger, A., Newmarch, I., Cottman-Fields, M., Wimmer, J., Towsey, M., Zhang, J., & Roe, P. (2013). *Reconciling Folksonomic Tagging with Taxa for Bioacoustic Annotations*. Paper presented at the 14th International Conference on Web Information System Engineering (WISE 2013), Nanjing, China. http://dx.doi.org/10.1007/978-3-642-41230-1_25

Truskinger, A., Yang, H. F., Wimmer, J., Zhang, J., Williamson, I., & Roe, P. (2011). *Large Scale Participatory Acoustic Sensor Data Analysis: Tools and Reputation Models to Enhance Effectiveness*. Paper presented at the 2011 IEEE 7th International Conference on E-Science (e-Science), Stockholm. http://dx.doi.org/10.1109/eScience.2011.29

Villanueva-Rivera, L. J., & Pijanowski, B. C. (2012). Pumilio: A Web-Based Management System for Ecological Recordings. *Bulletin of the Ecological Society of America, 93*(1), 71-81. doi: 10.1890/0012-9623-93.1.71

Wang, A. (2006). The Shazam music recognition service. *Commun. ACM, 49*(8), 44-48. doi: 10.1145/1145287.1145312

Wimmer, J., Towsey, M., Planitz, B., Williamson, I., & Roe, P. (2013). Analysing environmental acoustic data through collaboration and automation. *Future Generation Computer Systems, 29*(2), 560-568. doi: http://dx.doi.org/10.1016/j.future.2012.03.004

Wimmer, J., Towsey, M., Roe, P., & Williamson, I. (2013). Sampling environmental acoustic recordings to determine bird species richness. *Ecological applications*. doi: 10.1890/12-2088.1

Zhu, X., Vondrick, C., Ramanan, D., & Fowlkes, C. (2012). *Do We Need More Training Data or Better Models for Object Detection?* Paper presented at the BMVC, Surrey.